

Multilevel Evaluation of Multidimensional Integral Transforms with Asymptotically Smooth Kernels

E.H. van Brummelen

*Eindhoven University of Technology, Department of Mechanical Engineering, P.O. Box 513,
5600 MB Eindhoven, The Netherlands.*

and

C.H. Venner

*University of Twente, Faculty of Engineering Technology, P.O. Box 217, 7500 AE Enschede,
The Netherlands.*

E-mail: e.h.v.brummelen@tue.nl; c.h.venner@utwente.nl.

In many practical applications of numerical methods a substantial increase in efficiency can be obtained by using local grid refinement, since the solution is generally smooth in large parts of the domain and large gradients occur only locally. Fast evaluation of integral transforms on such an adaptive grid requires an algorithm that relies on the smoothness of the continuum kernel only, independent of its discrete form. A multilevel algorithm with this property was presented in [8, A. Brandt and C.H. Venner, SIAM J. Sci. Stat. Comput. **19** (1998) pp.468-492]. Ref. [8] shows that already on a uniform grid the new algorithm is more efficient than earlier fast evaluation algorithms, and elaborates the application to one-dimensional transforms. The present work analyses the extension and implementation of the algorithm for multidimensional transforms. The analysis conveys that the multidimensional extension is nontrivial, on account of the occurrence of nonlocal corrections. However, by virtue of the asymptotic smoothness properties of the continuum kernel, these corrections can again be evaluated fast. By recursion, it is then possible to obtain the optimal work estimates indicated in [8]. Currently, only uniform grids are considered. Detailed numerical results will be presented for a two dimensional model problem. The results demonstrate that with the new algorithm the evaluation of multidimensional transforms is also more efficient than with previous algorithms.

Key Words: multigrid, integral transform, singular smooth kernel, fast evaluation, local grid refinement

1. INTRODUCTION

In many fields in mathematics, physics and engineering, the numerical evaluation of integral transforms or multi-integrals of the type:

$$Gu(\mathbf{x}) = \int_{\Omega} G(\mathbf{x}, \mathbf{y}) u(\mathbf{y}) d\mathbf{y}, \quad \mathbf{x} \in \underline{\Omega} \subset \mathbb{R}^d, \mathbf{y} \in \Omega \subset \mathbb{R}^d, \quad (1)$$

is a frequently arising task, e.g., in elasticity-problems, integro-differential equations, integral equations, astrophysics and computer graphics. The evaluation of (1) can be a task by itself or a subtask in the solution of a (system of) integro-differential equation(s). In the latter case, $u(\mathbf{y})$ is the unknown function.

To evaluate (1) numerically, the continuous transform is replaced by a matrix multiplication or “multisummation”, i.e. at the expense of a discretization error the evaluation of (1) is replaced by the n -vector $\mathbf{Gu} = \mathbf{G} \cdot \mathbf{u}$, given the $\bar{n} \times n$ dense matrix \mathbf{G} and the \bar{n} -vector \mathbf{u} . Multisummations of this form also appear in, for instance, particle physics (Coulombic molecular interaction).

Straightforward evaluation of the matrix-vector product $\mathbf{G} \cdot \mathbf{u}$ involves $\bar{n}n$ operations. If the matrix \mathbf{G} has arbitrary entries, no faster method than straightforward multiplication exists. However, many cases arise in which the “discrete kernel” \mathbf{G} has special properties that can be used to obtain a fast evaluation algorithm. Several approaches have been suggested to reduce the computational cost of the multisummation to below $\bar{n}n$ operations, by exploiting such special properties, e.g., hierarchical solvers for many body interaction problems [1, 3], multipole expansions [11], Fast Fourier Transform based schemes [12] and wavelet techniques.

In [6], a general approach referred to as *multilevel matrix multiplication* or *multilevel multi-integration* was presented. The algorithm has been applied to, for instance, integral transforms in elastohydrodynamic lubrication problems [13] and, more recently, to integral transforms emanating from discretizations of the Laplace and Helmholtz equations by the boundary-element method [10, 9]. The algorithm in [6] relies on the smoothness of the matrix \mathbf{G} . For particle problems $G_{ij} = G(\mathbf{x}_i, \mathbf{y}_j)$ and the smoothness of the discrete kernel follows immediately from the smoothness of the continuum kernel. However, the discretization of (1) presented in [6] yields a matrix of which the smoothness is not only determined by the smoothness of the continuum kernel, but also by the applied grid. The fast evaluation algorithm then requires grid uniformity.

On the other hand, in practical applications, e.g., in contact mechanics and in lubrication, a substantial increase in efficiency can be obtained by employing non-uniform grids, since the solution is often smooth in large parts of the domain and large gradients occur only locally. Moreover, if $u(\mathbf{y})$ has some singularity, local grid refinement is even imperative to maintain an efficient work to accuracy relationship.

The multilevel methodology in principle allows local grid refinements in a very natural way, see [2, 4, 5], but to implement these techniques for integral transforms a new algorithm had to be developed. This new algorithm was presented in [8]. For its efficiency, the algorithm relies exclusively on the smoothness of the *continuum* kernel, thereby allowing the use of local grid refinements and grid adaptivity. In [8], it was tested for a one dimensional problem on a uniform grid and it was shown that already on a uniform grid the evaluation is more efficient than with previous

algorithms. The application to an actual one dimensional problem where local refinement is essential to maintain optimal efficiency was discussed in [7].

In the present work, the extension and implementation of the algorithm for higher dimensional transforms is discussed. To separate the complications of grid non-uniformity and multiple dimensions, only uniform grids are considered. The implementation with locally refined grids is deferred to future research. Numerical results are presented for a two dimensional model problem.

2. DISCRETIZATION

In this section we briefly review the discretization procedure for (1). The details of the procedure can be found in [8]. Throughout, it will be assumed that $\underline{d} = d$. The generalization to more general cases is straightforward.

The domain Ω is divided into subdomains $\Omega_j^h = \{\mathbf{y} \in \mathbb{R}^d \mid y_{k_{j_k}} \leq y_k \leq y_{k_{j_k+1}}, 1 \leq k \leq d\}$. The resulting grid, $\{\mathbf{y}_j\}$, is referred to as the integration grid. The integral (1) can now be rewritten as a summation of the contributions of the individual subdomains, defined by:

$$G_j^h u(\mathbf{x}) = \int_{\Omega_j^h} G(\mathbf{x}, \mathbf{y}) u(\mathbf{y}) d\mathbf{y}. \quad (2)$$

Next, let $G^l(\mathbf{x}, \mathbf{y})$ be a family of kernels, defined recursively,

$$\begin{aligned} G^0(\mathbf{x}, \mathbf{y}) &= G(\mathbf{x}, \mathbf{y}), \\ G^l(\mathbf{x}, \mathbf{y}) &= \int_{x_k}^{y_k} G^{l-\mathbf{e}_k}(\mathbf{x}, \mathbf{y} + (\eta - y_k) \mathbf{e}_k) d\eta, \end{aligned} \quad (3)$$

where \mathbf{e}_k denotes the k -th unit vector. Note that $\mathbf{y} - y_k \mathbf{e}_k$ is \mathbf{y} with its k -th component set to zero. In many practical cases, e.g., for the logarithmic kernel in [8] and for the kernel in our model problem,

$$G(\mathbf{x}, \mathbf{y}) = |\mathbf{y} - \mathbf{x}|^{-1}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^2, \quad (4)$$

it is possible to derive $G^l(\mathbf{x}, \mathbf{y})$ analytically.

The function $u(\mathbf{y})$ is approximated on Ω_j^h by $\tilde{u}_j^h(\mathbf{y})$, an order $\mathbf{s} = (s_1, \dots, s_d)$ interpolation polynomial, i.e. a polynomial of degree $\mathbf{s} - 1$. The interpolation is done from a data-grid of points, $\{\mathbf{z}_j\}$, on which for every site $u_j^h = u(\mathbf{z}_j^h)$ is given. For smallest errors, the integration interval should be central relative to the interpolation points. However, near external boundaries this may no longer be possible.

A discrete approximation to (2) is obtained by replacing $u(\mathbf{y})$ by $\tilde{u}^h(\mathbf{y})$ and integrating by parts \mathbf{s} times:

$$G_j^h \tilde{u}^h(\mathbf{x}) = \sum_{l=1}^{\mathbf{s}} \sum_{\mathbf{a}=0}^1 (-1)^{||+|\mathbf{a}|} G^l(\mathbf{x}, \mathbf{y}_{j+\mathbf{a}}) \tilde{u}_j^{h, (l-1)}(\mathbf{y}_{j+\mathbf{a}}), \quad (5)$$

where $|| = \sum_{k=1}^d l_k$, $|\mathbf{a}| = \sum_{k=1}^d a_k$ and $\tilde{u}_j^{h, (l-1)}(\mathbf{y})$ denotes the $l_k - 1$ derivative of $\tilde{u}_j^h(\mathbf{y})$ to y_k for all k . Note that summation over a vector implies summation

over each of the components of the vector, so that the summation in (5) actually extends over all vertices of the subdomain Ω_j^h .

The integral transform (1) can now be approximated by taking the sum of (5) over all subdomains Ω_j^h . Rewriting this summation, we obtain a sum of $\prod_{k=1}^d s_k$ discrete subtransforms, $S^{h,l}(\mathbf{x})$, and boundary terms, $B^{h,l}(\mathbf{x})$:

$$G^h \tilde{u}^h(\mathbf{x}) = \sum_{l=1}^s (-1)^{d+|l|} B^{h,l}(\mathbf{x}) + \sum_{l=1}^s (-1)^{d+|l|} S^{h,l}(\mathbf{x}), \quad (6)$$

with the discrete subtransforms $S^{h,l}(\mathbf{x})$ defined by:

$$S^{h,l}(\mathbf{x}) = \sum_{j=0}^n G^l(\mathbf{x}, y_j) U_j^{h,l}, \quad (7)$$

where

$$U_j^{h,l} = \begin{cases} \sum_{a=0}^1 (-1)^{d+|a|} \tilde{u}_{j-a}^{h,(l-1)}(y_j), & \forall k \left(\frac{1}{2}s_k \leq j_k \leq n_k - \frac{1}{2}s_k \right), \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

The boundary terms, $B^{h,l}(\mathbf{x})$, extend over all nodes bounding the domain Ω and the subdomains Ω_j^h where the integration interval is not central relative to the interpolation points:

$$B^{h,l}(\mathbf{x}) = \sum_{j=0}^n G^l(\mathbf{x}, y_j) V_j^{h,l}, \quad (9)$$

where

$$V_j^{h,l} = \begin{cases} \sum_{a=0}^1 (-1)^{d+|a|} \tilde{v}_{j-a}^{h,(l-1)}(y_{j-a}), & \exists k \left(0 \leq j_k < \frac{1}{2}s_k \vee n_k - \frac{1}{2}s_k < j_k \leq n_k \right), \\ 0, & \text{otherwise.} \end{cases}$$

with

$$\tilde{v}_j^{h,(l-1)}(y_j) = \begin{cases} \tilde{u}_j^{h,(l-1)}(y_j), & \forall k \left(0 \leq j_k < n_k \right), \\ 0, & \text{otherwise.} \end{cases}$$

Notice that by (7), the kernel in each of the transforms follows from the continuum kernel $G(\mathbf{x}, y)$ by integration.

Assuming that $u(y)$ is s times differentiable on Ω , in the case of a uniform grid, the discretization error, i.e. the difference between (6) and (1), per unit of integration is bounded by

$$|G^h \tilde{u}^h(\mathbf{x}) - Gu(\mathbf{x})| \leq \alpha_1 \sum_{k=1}^d (\gamma_1 h_k)^{s_k} \|u^{(s_k \mathbf{e}_k)}\|_{\max, \Omega} \|G\|_{1, \Omega}, \quad (10)$$

with h_k the mesh size of $\{y_j\}$ in the k -direction, $\|u^{(s_k \mathbf{e}_k)}\|_{\max, \Omega}$ the maximum of the s_k derivative of $u(y)$ to y_k on Ω and $\|G\|_{1, \Omega}$ the average of $|G(\mathbf{x}, y)|$ over the

integration domain for a particular \mathbf{x} . Further, α_1 and γ_1 are constants, the latter depending on the interpolation geometry.

As a result of symmetry and anti-symmetry of the interpolation polynomials, on a uniform grid many of the derivatives of the approximating functions, $\tilde{u}_j^h(\mathbf{y})$, are continuous across the subdomain boundaries and, consequently, the corresponding $U_j^{h,l}$ vanish for all j . In particular, $U_j^{h,l} = 0$ for any l_k odd and s_k even, i.e. if the integration grid coincides with the data-grid ($y_{k,j_k} = z_{k,j_k}$), and for any l_k even and s_k odd, if the integration grid coincides with data-grid midpoints ($y_{k,j_k} = (z_{k,j_k} + z_{k,j_k-1})/2$); see [8]. Hence, the number of transforms that actually need to be evaluated is $\prod_{k=1}^d (\bar{s}_k/2)$, where $\bar{s}_k = s_k$ if s_k is even and $\bar{s}_k = s_k + 1$ if s_k is odd.

To illustrate the discretization procedure, consider the discretization (6) with kernel (4) and $\tilde{u}_j^h(\mathbf{y})$ a bi-linear interpolation from the data-grid ($\mathbf{s} = 2$). The integration grid and data-grid coincide, as is usual for \mathbf{s} is even. It is easily verified that indeed $U_j^{h,l}$ vanishes if either $l_1 = 1$ or $l_2 = 1$. Hence, only the discrete transform $S^{h,(2,2)}(\mathbf{x})$ needs to be evaluated. Introducing $\mathbf{t} = \mathbf{y} - \mathbf{x}$, the integrated kernel for the transform reads:

$$G^{(2,2)}(\mathbf{x}, \mathbf{y}) = \frac{1}{2} t_1 |t_1| t_2 \operatorname{arcsinh} \left(\frac{t_2}{t_1} \right) + \frac{1}{2} t_2 |t_2| t_1 \operatorname{arcsinh} \left(\frac{t_1}{t_2} \right) + \frac{1}{6} \left(|t_1|^3 + |t_2|^3 - (t_1^2 + t_2^2)^{3/2} \right). \quad (11)$$

Inserting the bi-linear approximation into (7) and (8) yields:

$$S^{h,(2,2)}(\mathbf{x}) = \sum_{j=1}^{n-1} G^{(2,2)}(\mathbf{x}, \mathbf{y}_j) U_j^{h,(2,2)}, \quad (12)$$

with the stencil of $U_j^{h,(2,2)}$ (for stencil notation see, e.g., [14])

$$U_j^{h,(2,2)} = \frac{1}{h_1 h_2} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}. \quad (13)$$

Section 6 presents numerical results using this discretization.

3. FAST EVALUATION OF DISCRETE TRANSFORMS

Consider the evaluation of the grid h discrete (sub)transform: $\forall \mathbf{x} \in \{\mathbf{x}_i\}$,

$$S^{h,l}(\mathbf{x}) = \sum_{j=0}^n G^l(\mathbf{x}, \mathbf{y}_j) U_j^{h,l}. \quad (14)$$

The kernel $G^l(\mathbf{x}, \mathbf{y})$ is assumed to be asymptotically smooth in x_k and y_k . This implies that $G^l(\mathbf{x}, \mathbf{y})$ is increasingly smooth in x_k and y_k , so that for all allowed errors, $\epsilon \in \mathbb{R}$, $\epsilon > 0$, and all scales $H \in \mathbb{R}$, $H > 0$, there exist $p, m \in \mathbb{N}$ for which a *softened kernel* $G_{He_k}^l(\mathbf{x}, \mathbf{y})$ can be defined with the properties:

- (i) *Locality*: $G_{He_k}^l(\mathbf{x}, \mathbf{y}) = G^l(\mathbf{x}, \mathbf{y})$ for $|y_k - x_k| \geq mH$.

(ii) $G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{y})$ is *suitably smooth* in x_k and y_k on the scale H .

Property (ii) means that, both as a function of x_k for any fixed $(\mathbf{x} - x_k\mathbf{e}_k, \mathbf{y})$ and as a function of y_k for any fixed $(x, \mathbf{y} - y_k\mathbf{e}_k)$, $G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{y})$ can be approximated up to an error of at most ϵ by a p -order interpolation from values $\{G_{H\mathbf{e}_k}^l(\mathbf{x} + (jh - x_k)\mathbf{e}_k, \mathbf{y}) \mid j \in \mathbb{Z}\}$ and $\{G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{y} + (jh - y_k)\mathbf{e}_k) \mid j \in \mathbb{Z}\}$, for any $h \in (0, H]$. The order of interpolation p is required to increase only moderately with decreasing ϵ , in particular, $p = O(\log 1/\epsilon)$ as $\epsilon \rightarrow 0$. Suitable smoothness of $G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{y})$ translates into the requirement that

$$(\gamma_2 H)^p |G_{H\mathbf{e}_k}^{l(p\mathbf{e}_k)}(\mathbf{x}, \mathbf{y})| \leq O(\epsilon), \quad (15)$$

for any (\mathbf{x}, \mathbf{y}) in the domain of interest. Here, $G_{H\mathbf{e}_k}^{l(p\mathbf{e}_k)}(\mathbf{x}, \mathbf{y})$ denotes a p -order derivative with respect to either x_k or y_k . Further, γ_2 is a constant depending on the interpolation geometry. In particular, $\gamma_2 = 1/2$ for the usual central interpolations.

The transform (14) can now be replaced by a softened transform and a correction:

$$S^{\mathbf{h}, \mathbf{l}}(\mathbf{x}) = S_{H\mathbf{e}_k}^{\mathbf{h}, \mathbf{l}}(\mathbf{x}) + M_{H\mathbf{e}_k}^{\mathbf{h}, \mathbf{l}}(\mathbf{x}), \quad (16)$$

with

$$S_{H\mathbf{e}_k}^{\mathbf{h}, \mathbf{l}}(\mathbf{x}) = \sum_{j=0}^n G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{y}_j) U_j^{\mathbf{h}, \mathbf{l}} \quad (17)$$

and

$$M_{H\mathbf{e}_k}^{\mathbf{h}, \mathbf{l}}(\mathbf{x}) = \sum_{j=0}^n (G^l(\mathbf{x}, \mathbf{y}_j) - G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{y}_j)) U_j^{\mathbf{h}, \mathbf{l}}. \quad (18)$$

Note that by (i), the correction (18) is local in the k -direction and only involves points with $|y_{kj} - x_k| < mH$.

Exploiting the suitable smoothness of the softened kernel, at the expense of an error $O(\epsilon)$, one may replace $G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{y}_j)$ by a p -order interpolation from its values on a grid $\{\mathbf{Y}_J\}$ with mesh sizes $\mathbf{H} = \mathbf{h} + (H - h_k)\mathbf{e}_k$. Specifically, there are interpolation weights $w_{jJ}^{\mathbf{h}, \mathbf{H}}$ such that for all \mathbf{x} :

$$S_{H\mathbf{e}_k}^{\mathbf{h}, \mathbf{l}}(\mathbf{x}) = \sum_{j=0}^n \sum_{J \in \Gamma_j^p} w_{jJ}^{\mathbf{h}, \mathbf{H}} G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{Y}_J) U_j^{\mathbf{h}, \mathbf{l}} + O(\epsilon). \quad (19)$$

where Γ_j^p stands for a set of p nodes in the neighborhood of \mathbf{y}_j , e.g., for central p -order interpolations, $\Gamma_j^p = \{J \in \mathbb{Z} \mid |y_{ij} - Y_{iJ}| \leq \delta_{ik} p H / 2\}$, with δ_{ik} the Kronecker delta. Changing the order of summation in (19) and neglecting $O(\epsilon)$ errors:

$$S_{H\mathbf{e}_k}^{\mathbf{h}, \mathbf{l}}(\mathbf{x}) = \sum_{J=0}^N G_{H\mathbf{e}_k}^l(\mathbf{x}, \mathbf{Y}_J) U_J^{\mathbf{H}, \mathbf{l}}, \quad (20)$$

with

$$U_J^{\mathbf{H}, \mathbf{l}} = \sum_{j \in \Gamma_J^p} w_{jJ}^{\mathbf{h}, \mathbf{H}} U_j^{\mathbf{h}, \mathbf{l}}. \quad (21)$$

The operation (21) is commonly referred to as antepolation, since it is the *adjoint* of *interpolation*.

Next, let $\{X_I\}$ denote an evaluation grid with mesh sizes H . By the smoothness of $G_{He_k}^l(x, y)$ with respect to x_k on the scale H , for any x_i there are interpolation weights \bar{w}_{il}^{hH} such that for all y :

$$G_{He_k}^l(x_i, y) = \sum_{l \in \Gamma_i^p} \bar{w}_{il}^{hH} G_{He_k}^l(X_I, y) + O(\epsilon) \quad (22)$$

From (22) it follows that, neglecting $O(\epsilon)$ errors:

$$S_{He_k}^{h,l}(x_i) = \sum_{l \in \Gamma_i^p} \bar{w}_{il}^{hH} S_{He_k}^{H,l}(X_I), \quad (23)$$

with $S_{He_k}^{H,l}$ denoting a grid H transform defined as:

$$S_{He_k}^{H,l}(X_I) = \sum_{J=0}^N G_{He_k}^l(X_I, Y_J) U_J^{H,l} \quad (24)$$

Summarizing, by (20) to (24), at the expense of an error $O(\epsilon)$, the grid h evaluation can be replaced by:

- (i) *antepolation* of $U_j^{h,l}$ from the integration grid $\{y_j\}$ to the coarse integration grid, $\{Y_J\}$, by (21).
- (ii) *evaluation* of $S_{He_k}^{H,l}$ on grid H .
- (iii) *interpolation* of $S_{He_k}^{H,l}$ from the coarse evaluation grid, $\{X_I\}$, to the evaluation grid $\{x_i\}$.
- (iv) addition of the local *correction* $M_{He_k}^{h,l}$, for all points of $\{x_i\}$.

Denoting by n the number of point on grid h , the cost of the transfer operations (antepolation and interpolation) is $O(pn)$. The corrections (18) arise in regions where the kernel is insufficiently smooth to be accurately approximated by a p -order interpolation from a grid with mesh size H . These regions are of dimension $d-1$ and the work invested in the corrections is $O(mn^{2-1/d})$. It is important to notice that for $d = 1$, the grid h evaluation can be transfered to grid H by $O(n)$ operations. Of course, the coarse grid evaluation (ii) can again be replaced by successive (i)-(iv) to transfer the multisummation to an even coarser grid. Hence, the process can be repeated recursively until a grid is reached at which the evaluation can be performed in $O(n)$ operations by direct summation. The grid h multisummation can thus be evaluated with asymptotically optimal efficiency, i.e. in $O(n)$ operations.

If $d \geq 2$, however, straightforward evaluation of the corrections inhibits optimal efficiency. To recover optimal efficiency, it is necessary to reduce the cost of the corrections to $O(n)$ operations. For this purpose, the correction (18) is rewritten as

$$\begin{aligned} M_{He_k}^{h,l}(x) &= \sum_{j=0}^n \left(G_{He_k}^l(x, y_j) - G_{H(e_k+e_q)}^l(x, y_j) \right) U_j^{h,l} \\ &+ \sum_{j=0}^n \left((G^l(x, y_j) - G_{He_k}^l(x, y_j)) - (G_{He_k}^l(x, y_j) - G_{H(e_k+e_q)}^l(x, y_j)) \right) U_j^{h,l}, \quad (25) \end{aligned}$$

with $q \neq k$. Assuming that a softened kernel inherits its asymptotic smoothness properties from the original kernel, the softened kernel $G_{H(\mathbf{e}_k + \mathbf{e}_q)}^l(\mathbf{x}, \mathbf{y}_j)$ can be constructed such that it is suitably smooth in x_k, x_q, y_k and y_q on the scale H and that the second multisummation in (25) is local in the k - and q -directions. By the suitable smoothness of the softened kernel in x_q and y_q on the scale H , the first multisummation in (25) can be transferred to a grid that is coarse in the q -direction. The process of separating a correction into a softened correction, that can be transferred to a coarser grid, and a lower-dimensional correction, can be repeated recursively with respect to all coordinate directions. The grid h corrections can then be evaluated in $O(n)$ operations.

Separating the transform (24) as

$$S_{H\mathbf{e}_k}^{H,l}(\mathbf{X}_I) = \sum_{J=0}^N G_{H(\mathbf{e}_k + \mathbf{e}_q)}^l(\mathbf{X}_I, \mathbf{Y}_J) U_J^{H,l} + \sum_{J=0}^N (G_{H\mathbf{e}_k}^l(\mathbf{X}_I, \mathbf{Y}_J) - G_{H(\mathbf{e}_k + \mathbf{e}_q)}^l(\mathbf{X}_I, \mathbf{Y}_J)) U_J^{H,l}, \quad (26)$$

it is evident that the multisummation (24) can be transferred to a grid that is coarse in the q -direction at the expense of a correction that is local in the q -direction. Of course, this process can also be repeated recursively with respect to all coordinate directions.

Summarizing, to evaluate the grid h discrete (sub)transform (14) fast, the operations (i)-(iv) are recursively applied to transfer the multisummation to grids that are increasingly coarse in each direction, until a grid is reached at which the multisummation can be performed in $O(n)$ operations by direct summation. All corrections that arise are treated in the same manner. The treatment of the corrections ensures that the correctional work is $O(1)$ operations per grid point (of the grid on which the corrections are required). On sufficiently fine grids the corrections are negligible compared to the discretization error which is made anyway, so that corrections can be avoided at all, i.e., $m = 0$ can be used on the finest grids; see [9] and appendix A. However, at this point it is noted that this only applies to the magnitude of the evaluation error. If the evaluation with a certain accuracy is not a final goal, but a subtask in the numerical solution of the integral equation, then a minimum softening distance $m = O(p)$ is needed to ensure that the fast evaluation operator has the same stability properties as the unigrid evaluation operator for highly oscillatory components. If one only considers the evaluation of the integral transform, however, corrections on the finest grids are unnecessary. For large scale computations the work involved in the fast evaluation is then only determined by the costs of the transfers on the finer grids, and the additional cost of the coarsest grid multisummation. The work estimates for the evaluation of all discrete subtransforms as indicated in [8] can then indeed be obtained.

Although it is most efficient to apply the softening and coarsening per direction, it is usually more convenient to first soften the kernel with respect to all coordinate directions and then transfer the multisummation. The additional expenses are only marginal.

4. KERNEL SOFTENING

In the previous section, we showed that the discrete subtransforms $S^{h,l}(\mathbf{x})$, resulting from the discretization of (1), in principle can be evaluated fast by separating each of the transforms in a softened transform and a local correction. The multisummation that is required to evaluate the softened transform can then be transferred to a coarser grid. As a result of the suitable smoothness of the *softened kernel* on the coarse grid scale, the evaluation error thus introduced is less than the fine grid discretization error.

In [8] it was shown that for 1-dimensional kernels a convenient softening can be obtained by locally replacing the original kernel with a polynomial, $P_H(\mathbf{x}, \mathbf{y}) = \sum_{i=0}^{2p-1} a_i(\mathbf{y} - \mathbf{x})^i$, in such a manner that the resulting kernel is $p-1$ times continuously differentiable. This approach can be extended to multidimensional kernels, by allowing the polynomial coefficients to depend on a reduced set of variables. In particular, for properly chosen softening distance m and softening order p ,

$$G_{H\mathbf{e}_k}^l(\mathbf{t}) = \begin{cases} P_{H\mathbf{e}_k}^l(\mathbf{t}) \equiv \sum_{i=0}^{2p-1} a_i(\mathbf{t} - t_k \mathbf{e}_k) t_k^i, & |t_k| \leq mH \\ G^l(\mathbf{t}), & \text{otherwise,} \end{cases} \quad (27)$$

with $\mathbf{t} = \mathbf{y} - \mathbf{x}$, defines a softened kernel that is suitably smooth in t_k on the scale H , provided that the coefficients, $a_i(\mathbf{t} - t_k \mathbf{e}_k)$, satisfy the continuity conditions

$$\sum_{i=j}^{2p-1} a_i(\mathbf{t} - t_k \mathbf{e}_k) \frac{i!}{(i-j)!} (\pm mH)^{i-j} = G^{l-j\mathbf{e}_k}(\mathbf{t} - (t_k \pm mH)\mathbf{e}_k), \quad j = 0, \dots, p-1. \quad (28)$$

The $2p$ coefficients $a_i(\mathbf{t} - t_k \mathbf{e}_k)$ in equation (27) are uniquely determined by the $2p$ continuity conditions (28). One may note that by (27), the operation is local in t_k . Commonly, $G^l(\mathbf{t})$ is either an even or an odd function of t_k and $a_i(\mathbf{t} - t_k \mathbf{e}_k) = 0$ for all odd i or all even i , respectively. Moreover, one should anticipate that usually the softening polynomial can be condensed to a convenient form that can be evaluated efficiently.

By (28), the polynomial coefficients, $a_i(\mathbf{t} - t_k \mathbf{e}_k)$, are a linear combination of the kernel derivatives $G^{l-j\mathbf{e}_k}(\mathbf{t} - (t_k \pm mH)\mathbf{e}_k)$, $j = 0, \dots, p-1$. Therefore, if the kernel $G^l(\mathbf{t})$ consists of a summation of components, then each of these components can be softened independently to form the softened kernel. Moreover, the asymptotic smoothness properties of the original kernel are maintained during the softening operation. Hence, if the original kernel is asymptotically smooth in t_q ($q \neq k$), then $G_{H\mathbf{e}_k}^l(\mathbf{t})$ can be softened in the q -direction to create a kernel that is suitably smooth in t_k and t_q . The resulting kernel again inherits its asymptotic smoothness properties from the original kernel. Consequently, if the original kernel is asymptotically smooth in \mathbf{t} , then sequential application of the softening operation with respect to each coordinate direction yields a softened kernel that is suitably smooth in \mathbf{t} .

To illustrate the multidimensional softening procedure, we consider the softening of the kernel (11). Notice that the kernel consists of a sum of components:

$G^{(2,2)}(\mathbf{t}) = \sum_{i=0}^{i=5} G_i^{(2,2)}(\mathbf{t})$, with

$$\begin{aligned} G_0^{(2,2)}(\mathbf{t}) &= \frac{1}{2} t_1 |t_1| t_2 \operatorname{arcsinh} \left(\frac{t_2}{t_1} \right), & G_3^{(2,2)}(\mathbf{t}) &= \frac{1}{2} t_2 |t_2| t_1 \operatorname{arcsinh} \left(\frac{t_1}{t_2} \right), \\ G_1^{(2,2)}(\mathbf{t}) &= -\frac{1}{6} t_1^2 \sqrt{t_1^2 + t_2^2}, & G_4^{(2,2)}(\mathbf{t}) &= -\frac{1}{6} t_2^2 \sqrt{t_1^2 + t_2^2}, \\ G_2^{(2,2)}(\mathbf{t}) &= \frac{1}{6} |t_1^3|, & G_5^{(2,2)}(\mathbf{t}) &= \frac{1}{6} |t_2^3|. \end{aligned} \quad (29)$$

The component $G_5^{(2,2)}(\mathbf{t})$ requires no softening with respect to t_1 since it is already sufficiently smooth. Assuming that identical softening parameters (m and p) are chosen in both coordinate directions, in regions where softening with respect to t_1 is required, i.e. for $|t_1| \leq mH$, the softening polynomials read

$$P_{0He_1}^{(2,2)}(\mathbf{t}) = \frac{1}{2} t_1^2 t_2 \operatorname{arcsinh} \left(\frac{t_2}{mH} \right) \quad (30)$$

$$+ \frac{(mH)^{2p}}{\left((mH)^2 + t_2^2\right)^{p-\frac{3}{2}}} \sum_{i=1}^{p-1} \sum_{j=1}^p A_{ij}^{(0)} \left(\frac{t_1}{mH} \right)^{2i} \left(\frac{t_2}{mH} \right)^{2j} \quad (31)$$

$$P_{1He_1}^{(2,2)}(\mathbf{t}) = \frac{(mH)^{2p}}{\left((mH)^2 + t_2^2\right)^{p-\frac{3}{2}}} \sum_{i=1}^{p-1} \sum_{j=1}^p A_{ij}^{(1)} \left(\frac{t_1}{mH} \right)^{2i} \left(\frac{t_2}{mH} \right)^{2j} \quad (32)$$

$$P_{2He_1}^{(2,2)}(\mathbf{t}) = (mH)^3 \sum_{i=1}^{p-1} A_i^{(2)} \left(\frac{t_1}{mH} \right)^{2i} \quad (33)$$

$$\begin{aligned} P_{3He_1}^{(2,2)}(\mathbf{t}) &= \frac{(mH)^{2p}}{\left((mH)^2 + t_2^2\right)^{p-\frac{3}{2}}} \sum_{i=1}^{p-1} \sum_{j=1}^p A_{ij}^{(3)} \left(\frac{t_1}{mH} \right)^{2i} \left(\frac{t_2}{mH} \right)^{2j} \\ &+ \operatorname{arcsinh} \left| \frac{mH}{t_2} \right| \frac{(mH)^{2p-1}}{\left((mH)^2 + t_2^2\right)^{p-2}} \sum_{i=1}^{p-1} \sum_{j=1}^p B_{ij}^{(3)} \left(\frac{t_1}{mH} \right)^{2i} \left(\frac{t_2}{mH} \right)^{2j} \end{aligned} \quad (34)$$

$$P_{4He_1}^{(2,2)}(\mathbf{t}) = \frac{(mH)^{2p}}{\left((mH)^2 + t_2^2\right)^{p-\frac{3}{2}}} \sum_{i=1}^{p-1} \sum_{j=1}^p A_{ij}^{(4)} \left(\frac{t_1}{mH} \right)^{2i} \left(\frac{t_2}{mH} \right)^{2j} \quad (35)$$

As an example, the coefficients in equations (31) to (35) are listed in Table 1 for softening-order $p = 4$. Note that by (27), $G_{iHe_1}^{(2,2)}(\mathbf{t}) = G_i^{(2,2)}(\mathbf{t})$ if $|t_1| > mH$.

To obtain a softened kernel that is smooth in both t_1 and t_2 , subsequently, the kernel is softened with respect to t_2 . Because the original kernel has the symmetry property $G^{(2,2)}(t_1, t_2) = G^{(2,2)}(t_2, t_1)$, equations (31) to (35) determine the softened kernel in regions where the original kernel is suitably smooth in either t_1 or t_2 . To obtain the softened kernel in the region where the original kernel is unsmooth in both t_1 and t_2 , the softening operation with respect to t_2 is applied to the softened

TABLE 1
Coefficients in equations (31) to (36) for softening-order $p = 4$.

$A^{(0)} = \begin{pmatrix} 0 & \frac{5}{32} & \frac{5}{24} & \frac{1}{12} & 0 \\ 0 & -\frac{1}{32} & \frac{3}{16} & \frac{1}{8} & 0 \\ 0 & -\frac{5}{32} & -\frac{1}{2} & -\frac{1}{4} & 0 \\ 0 & \frac{1}{32} & \frac{5}{48} & \frac{1}{24} & 0 \end{pmatrix}$	$A^{(1)} = \begin{pmatrix} \frac{1}{96} & 0 & 0 & 0 & 0 \\ -\frac{3}{32} & -\frac{5}{16} & -\frac{5}{12} & -\frac{1}{6} & 0 \\ -\frac{3}{32} & -\frac{5}{24} & -\frac{1}{12} & 0 & 0 \\ \frac{1}{96} & \frac{1}{48} & 0 & 0 & 0 \end{pmatrix}$	$A^{(2)} = \begin{pmatrix} -\frac{1}{96} & \frac{3}{32} & \frac{3}{32} & -\frac{1}{96} \end{pmatrix}$
$A^{(3)} = \begin{pmatrix} 0 & -\frac{23}{96} & -\frac{35}{96} & -\frac{5}{32} & 0 \\ 0 & \frac{7}{32} & \frac{5}{32} & \frac{1}{32} & 0 \\ 0 & \frac{1}{32} & \frac{9}{32} & \frac{5}{32} & 0 \\ 0 & -\frac{1}{96} & -\frac{7}{96} & -\frac{1}{32} & 0 \end{pmatrix}$	$B^{(3)} = \begin{pmatrix} 0 & \frac{5}{32} & \frac{5}{16} & \frac{5}{32} & 0 \\ 0 & \frac{15}{32} & \frac{15}{16} & \frac{15}{32} & 0 \\ 0 & -\frac{5}{32} & -\frac{5}{16} & -\frac{5}{32} & 0 \\ 0 & \frac{1}{32} & \frac{1}{16} & \frac{1}{32} & 0 \end{pmatrix}$	$A^{(4)} = \begin{pmatrix} 0 & -\frac{5}{96} & -\frac{5}{16} & -\frac{5}{12} & -\frac{1}{6} \\ 0 & -\frac{5}{32} & -\frac{5}{24} & -\frac{1}{12} & 0 \\ 0 & \frac{5}{96} & \frac{1}{48} & 0 & 0 \\ 0 & -\frac{1}{96} & 0 & 0 & 0 \end{pmatrix}$
$A^{(5)} = \begin{pmatrix} \frac{161\sqrt{2}}{6144} & \frac{\sqrt{2}(-671+480\sqrt{2}\ln(1+\sqrt{2}))}{6144} & -\frac{517\sqrt{2}}{6144} & \frac{67\sqrt{2}}{6144} \\ \frac{\sqrt{2}(-671+480\sqrt{2}\ln(1+\sqrt{2}))}{6144} & \frac{\sqrt{2}(321+2880\sqrt{2}\ln(1+\sqrt{2}))}{6144} & \frac{\sqrt{2}(-480\sqrt{2}\ln(1+\sqrt{2})-261)}{6144} & \frac{\sqrt{2}(96\sqrt{2}+35)}{6144} \\ -\frac{517\sqrt{2}}{6144} & \frac{\sqrt{2}(-480\sqrt{2}\ln(1+\sqrt{2})-261)}{6144} & \frac{83\sqrt{2}}{2048} & -\frac{47\sqrt{2}}{6144} \\ \frac{67\sqrt{2}}{6144} & \frac{\sqrt{2}(96\sqrt{2}+35)}{6144} & -\frac{47\sqrt{2}}{6144} & \frac{3\sqrt{2}}{2048} \end{pmatrix}$		

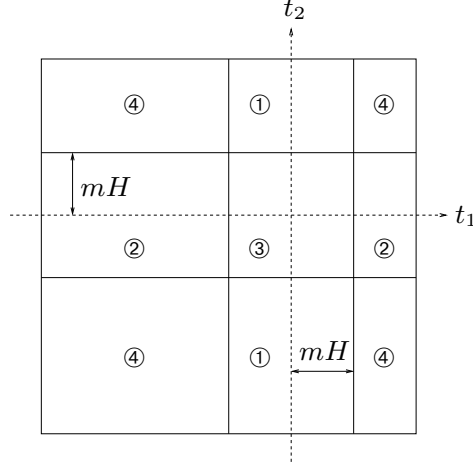


FIG. 1. Division of domain in softening regions: ①: softening in t_1 only, ②: softening in t_2 only, ③: softening in t_1 and t_2 , ④: original kernel is sufficiently smooth.

kernel $G_{H\mathbf{e}_1}^{(2,2)}(\mathbf{t})$. For $\mathbf{t} \in [-mH, mH]^2$, this yields the polynomial

$$P_{(H,H)}^{(2,2)}(\mathbf{t}) = (mH)^3 \sum_{i=1}^{p-1} \sum_{j=1}^{p-1} A_{ij}^{(5)} \left(\frac{t_1}{mH} \right)^{2i} \left(\frac{t_2}{mH} \right)^{2j} + \sum_{i=1}^{p-1} A_i^{(2)} \left(\left(\frac{t_1}{mH} \right)^{2i} + \left(\frac{t_2}{mH} \right)^{2i} \right). \quad (36)$$

For $p = 4$ the coefficients in (36) are listed in Table 1. One may note that $A_{ij}^{(5)}$ is symmetric, so that the symmetry of the original kernel in t_1 and t_2 is maintained.

From equations (31) to (36) it follows that the softened kernel is given by

$$G_{(H,H)}^{(2,2)}(\mathbf{t}) = \begin{cases} G_5^{(2,2)}(t_1, t_2) + \sum_{i=0}^4 P_{iH\mathbf{e}_1}^{(2,2)}(t_1, t_2) & |t_1| \leq mH, |t_2| > mH \quad \textcircled{1} \\ G_5^{(2,2)}(t_2, t_1) + \sum_{i=0}^4 P_{iH\mathbf{e}_1}^{(2,2)}(t_2, t_1) & |t_1| > mH, |t_2| \leq mH \quad \textcircled{2} \\ P_{(H,H)}^{(2,2)}(\mathbf{t}) & |t_1| \leq mH, |t_2| \leq mH \quad \textcircled{3} \\ G^{2,2}(\mathbf{t}) & \text{otherwise.} \quad \textcircled{4} \end{cases} \quad (37)$$

The encircled numbers in (37) refer to Figure 1.

5. WORK MINIMIZATION AND ERROR CONTROL

The multilevel fast evaluation algorithm relies on the smoothness of the softened kernel. For the softened kernels resulting from the operation defined in Section 4, the smoothness depends on the softening order, p , and the softening distance, m . The work invested in the fast evaluation algorithm also depends on p and m . In this section we derive the m and p for which the computational work is minimized

subject to the condition that the incremental evaluation error does not exceed the discretization error.

To obtain the constraint for \mathbf{m} and \mathbf{p} , we analyze the additional evaluation error on the target grid, \mathbf{h} , due to transferring the evaluation of the softened transform from a grid with mesh sizes $\mathbf{H}/2$ to a grid with mesh sizes \mathbf{H} . This error results from transferring the evaluation from the integration grid $\{\mathbf{y}_j^{\mathbf{H}/2}\}$ to the grid $\{\mathbf{y}_j^{\mathbf{H}}\}$, i.e. from replacing the softened kernel $G_{\mathbf{H}}^l(\mathbf{x}, \mathbf{y})$ for fixed \mathbf{x} and as a function of \mathbf{y} by a \mathbf{p} -order interpolation from $\{\mathbf{y}_j^{\mathbf{H}}\}$, and from replacing the transforms on the evaluation grid $\{\mathbf{x}_i^{\mathbf{H}/2}\}$ by an interpolation of transforms on $\{\mathbf{x}_i^{\mathbf{H}}\}$, i.e. from replacing $G_{\mathbf{H}}^l(\mathbf{x}, \mathbf{y})$ for fixed \mathbf{y} and as a function of \mathbf{x} by a \mathbf{p} -order interpolation from $\{\mathbf{x}_i^{\mathbf{H}}\}$.

The additional evaluation error, $E(\mathbf{x}_i^{\mathbf{h}})$, is the sum of the local interpolation errors, that is, the difference between the actual kernel value and its interpolation from the grids $\{\mathbf{y}_j^{\mathbf{H}}\}$ and $\{\mathbf{x}_i^{\mathbf{H}}\}$, weighted by $U_j^{\mathbf{h},l}$:

$$E(\mathbf{x}_i^{\mathbf{h}}) = \sum_{j=0}^n \left(G_{\mathbf{H}}^l(\mathbf{x}_i^{\mathbf{h}}, \mathbf{y}_j^{\mathbf{h}}) - \sum_{\mathbf{j} \in \Gamma_{\mathbf{p}}^{\mathbf{H}}} w_{\mathbf{j}\mathbf{j}}^{\mathbf{h}\mathbf{H}} \sum_{\mathbf{l} \in \Gamma_{\mathbf{p}}^{\mathbf{H}}} \bar{w}_{\mathbf{l}\mathbf{l}}^{\mathbf{h}\mathbf{H}} G_{\mathbf{H}}^l(\mathbf{x}_i^{\mathbf{H}}, \mathbf{y}_j^{\mathbf{H}}) \right) U_j^{\mathbf{h},l}, \quad (38)$$

In regions where $u(\mathbf{y})$ is \mathbf{s} -times differentiable,

$$|U_j^{\mathbf{h},l}| \leq \|u^{(\mathbf{s})}\|_{\max, \Omega} \prod_{k=1}^d \left(2(\gamma_3 h_k)^{s_k - l_k + 1} + O(h_k^{s_k - l_k + 2}) \right), \quad (39)$$

with $\gamma_3 \approx 0.5$ for a uniform grid; see [8]. Denoting by $\Theta = \{\mathbf{y} - \mathbf{x} \mid \mathbf{x} \in \underline{\Omega}, \mathbf{y} \in \Omega\}$, it follows from (38) and (39) that the error per unit of integration caused by the transfer from grid $\mathbf{H}/2$ to grid \mathbf{H} is bounded by

$$|\varepsilon^{\mathbf{H}}| \leq \alpha_2 \|G_{\mathbf{H}}^l - \mathbb{I}_{\mathbf{H}}^{\mathbf{H}/2} G_{\mathbf{H}}^l\|_{1, \Theta} \|u^{(\mathbf{s})}\|_{\max, \Omega} \prod_{k=1}^d \gamma_3 (\gamma_3 h_k)^{s_k - l_k}, \quad (40)$$

where α_2 is some positive constant and $\|G_{\mathbf{H}}^l - \mathbb{I}_{\mathbf{H}}^{\mathbf{H}/2} G_{\mathbf{H}}^l\|_{1, \Theta}$ denotes the average in Θ of the absolute value of the \mathbf{p} -order interpolation error, introduced by replacing $G_{\mathbf{H}}^l(\mathbf{x}, \mathbf{y})$ for fixed \mathbf{x} by an interpolation from $\{\mathbf{y}_j^{\mathbf{H}}\}$ and for fixed \mathbf{y} by an interpolation from $\{\mathbf{x}_i^{\mathbf{H}}\}$. From (10) and (40) it follows that the incremental evaluation error is smaller than the fine grid discretization error if the following requirement is satisfied:

$$\|G_{\mathbf{H}}^l - \mathbb{I}_{\mathbf{H}}^{\mathbf{H}/2} G_{\mathbf{H}}^l\|_{1, \Theta} \prod_{k=1}^d \gamma_3 (\gamma_3 h_k)^{s_k - l_k} \leq (\alpha_1 / \alpha_2) \sum_{k=1}^d (\gamma_1 h_k)^{s_k} \|G^l\|_{1, \Theta} \frac{\|u^{(s_k e_k)}\|_{\max, \Omega}}{\|u^{(\mathbf{s})}\|_{\max, \Omega}}. \quad (41)$$

Notice that in the one dimensional case the derivatives of $u(\mathbf{y})$ in the right-hand side of (41) cancel. The relation between the evaluation error and the discretization error is then independent of $u(\mathbf{y})$.

The interpolation error is composed of the interpolation error per direction. Clearly, the requirement that the incremental evaluation error is smaller than the fine grid discretization error is satisfied if for every direction the contribution to the evaluation error is smaller than the contribution to the discretization error. Hence,

requirement (41) can be separated in the following requirement per direction:

$$\|G_{\mathbf{H}}^{\mathbf{l}} - \mathbb{I}_{\mathbf{H}}^{\mathbf{H}-H_k \mathbf{e}_k/2} G_{\mathbf{H}}^{\mathbf{l}}\|_{1,\Theta} \leq h_k^{s_k-d(s_k-l_k)} c_{G,u} c_{\mathbf{h}} c_{\gamma}, \quad (42)$$

where

$$c_{G,u} = \|G\|_{1,\Theta} \frac{\|u^{(s_k \mathbf{e}_k)}\|_{\max,\Omega}}{\|u^{(\mathbf{s})}\|_{\max,\Omega}}, \quad c_{\mathbf{h}} = \prod_{i=1}^d \frac{h_k^{s_k-l_k}}{h_i^{s_i-l_i}}, \quad c_{\gamma} = (\alpha_1/\alpha_2) \gamma_1^{s_k} \prod_{i=1}^d \gamma_3^{-(s_i-l_i+1)}.$$

The average interpolation error on the left-hand side of (42) depends on the properties of $G_{\mathbf{H}}^{\mathbf{l}}(\mathbf{x}, \mathbf{y})$, and thus on the choice of \mathbf{m} and \mathbf{p} . The specific dependence of (42) on \mathbf{m} and \mathbf{p} is derived in Appendix A.

The computational work per grid $\mathbf{H}/2$ node involved in transferring the evaluation of the discrete transform from grid $\mathbf{H}/2$ to grid \mathbf{H} is estimated

$$W^{\mathbf{H}} = O(2[1-2^{-d}]p_k + 4dm_k \bar{H}^{1-d}), \quad (43)$$

where \bar{H}^{1-d} estimates the cost of the $d-1$ dimensional summation associated with the correction. The estimate (43) assumes that p_k and m_k are independent of k . This work estimate is obtained as follows: defining an operation to be one multiplication and one addition, the number of operations involved in the \mathbf{p} -order anteprolation from a grid with mesh sizes $\mathbf{H}/2$ to a semi-coarse grid with mesh sizes $(\mathbf{H} + \mathbf{e}_k H_k)/2$ is $p_k/2$, since for half of the values the transfer is trivial. The number of nodes on this semi-coarse grid is approximately half the number of nodes on the $\mathbf{H}/2$ grid. Hence, the next step in the anteprolation is performed in $p_k/4$ operations. In general, the i^{th} step in the anteprolation takes $p_k/2^i$ operations. The number of operations resulting from the interpolation is obtained in a similar manner, so that the total amount of work invested in the transfer operations is approximately $2[1-2^{-d}]p_k$. The evaluation of the corrections (18) involves $4m_k$ summations over a $d-1$ dimensional domain per direction per grid $\mathbf{H}/2$ node. These summations are again evaluated fast, so that the cost per grid point is $O(1)$. One should note that an accurate estimate of the cost of the corrections is not essential, since $m_k = 0$ is employed on the finest grids; see [8] and Appendix A. Assuming that the dimensions of the domains $\underline{\Omega}$ and Ω are $O(1)$, we arrive at the total work estimate (43).

The optimal transfer parameters are the \mathbf{m} and \mathbf{p} that minimize (43) subject to (42). As an example, the optimization process for the fast evaluation of the model problem is presented in Appendix B.

6. NUMERICAL EXPERIMENTS

The algorithm is tested for the integral transform with kernel (4) on a domain $\Omega = [-1, 1]^2$ with

$$u(\mathbf{y}) = \begin{cases} \prod_{k=1}^2 \left[-\frac{1}{3} + \left(\frac{10 y_k}{9} \right)^2 - \frac{2}{3} \left| \frac{10 y_k}{9} \right|^3 \right], & \mathbf{y} \in [-9/10, 9/10]^2, \\ 0, & \text{otherwise.} \end{cases} \quad (44)$$

TABLE 2
Transferparameters p_k and m_k used in the evaluation of $S^{h,(2,2)}$.

	$K = 5$		$K = 6$		$K = 7$		$K = 8$		$K = 9$		$K = 10$		$K = 11$	
L	p_k	m_k	p_k	m_k	p_k	m_k	p_k	m_k	p_k	m_k	p_k	m_k	p_k	m_k
K-1	4	0	4	0	4	0	4	0	4	0	4	0	4	0
K-2	6	2	4	1	4	0	4	0	4	0	4	0	4	0
K-3			6	3	6	3	6	2	4	1	4	1	4	0
K-4					8	5	8	4	6	3	6	3	6	2
K-5							10	6	8	5	8	5	8	4
K-6									10	8	10	7	10	6

The above problem is solved numerically using an $s = 2$ discretization on uniform grids with mesh widths $h_k = 2^{1-K}$, $K = 5, 6, \dots, 11$. All boundary terms (9) vanish and only the discrete transform $S^{h,(2,2)}(x)$ by (12) requires evaluation.

To evaluate this transform fast, the softening of $G^{(2,2)}(x, y)$ presented in Section 4 is used. Details of the derivation of the optimal transfer parameters are presented in Appendix B. The parameters used in the computations are obtained as follows: first p_k^* is calculated using equation (B.11), with $l_k = 2$, $h_k = 2^{1-K}$, $H_k = 2^{1-L}$ and the constant c_a in equation (B.13) set to 0. Next, p_k is obtained from:

$$p_k = \begin{cases} 2 \lfloor p_k^*/2 + 1 \rfloor, & p_k^* \geq l_k + 2, \\ l_k + 2, & \text{otherwise,} \end{cases} \quad (45)$$

Finally, m_k is obtained from:

$$m_k = \begin{cases} \left\lfloor \frac{1}{2} + \frac{3}{32} \bar{H} \gamma_2^{-1} \chi(p_k - l_k - 1) \right\rfloor, & p_k^* \geq l_k + 2, \\ 0, & \text{otherwise,} \end{cases} \quad (46)$$

with \bar{H} set to $H_k/2$. The values of p_k and m_k thus obtained are listed in Table 2. The table confirms that $m_k = 0$ can indeed be used on several of the finer grids and that for larger K the number of such grids increases.

To monitor the accuracy of the fast evaluation in relation to the discretization error, the error ϵ_K^L is defined by the l_2 -norm of the difference between the exact solution and the numerical solution that is obtained on level K when direct multi-summation is performed on level L :

$$\epsilon_K^L = \left(\frac{1}{n} \sum_{i=0}^n |(G^h \tilde{u}^h)^{K,L}(x_i^h) - Gu(x_i^h)|^2 \right)^{1/2}. \quad (47)$$

One may note that ϵ_K^K is the l_2 -norm of the discretization error on the level K grid. Table 3 lists the errors obtained for the model-problem. As a side note, we mention that the corrections are evaluated by means of the multilevel matrix multiplication algorithm; see [6]. The leftmost column confirms $O(h^2)$ -convergence of the discretization error. The entries marked by asterisks denote the results for $L = K/2$. In this case, the grid on which direct multisummation is performed

TABLE 3
Error ϵ_K^L for the model problem.

K	$L = K$	$K - 1$	$K - 2$	$K - 3$	$K - 4$	$K - 5$	$K - 6$
5	$2.01 \cdot 10^{-4}$	$2.05 \cdot 10^{-4}$	$3.60 \cdot 10^{-4}$				
6	$5.17 \cdot 10^{-5}$	$4.95 \cdot 10^{-5}$	$1.02 \cdot 10^{-4}$	$*1.98 \cdot 10^{-4}$			
7	$1.31 \cdot 10^{-5}$	$1.27 \cdot 10^{-5}$	$1.08 \cdot 10^{-5}$	$1.84 \cdot 10^{-5}$	$3.18 \cdot 10^{-5}$		
8	$\approx 3.3 \cdot 10^{-6}$	$3.24 \cdot 10^{-6}$	$2.81 \cdot 10^{-6}$	$3.12 \cdot 10^{-6}$	$*4.18 \cdot 10^{-6}$	$6.64 \cdot 10^{-6}$	
9	$\approx 8.1 \cdot 10^{-7}$		$7.54 \cdot 10^{-7}$	$1.06 \cdot 10^{-6}$	$1.38 \cdot 10^{-6}$	$2.13 \cdot 10^{-6}$	$3.55 \cdot 10^{-6}$
10	$\approx 2.0 \cdot 10^{-7}$			$2.18 \cdot 10^{-7}$	$2.33 \cdot 10^{-7}$	$*2.47 \cdot 10^{-7}$	$3.21 \cdot 10^{-7}$
11	$\approx 5.0 \cdot 10^{-8}$				$4.23 \cdot 10^{-8}$	$4.73 \cdot 10^{-8}$	$5.28 \cdot 10^{-8}$

TABLE 4
Incremental error $I\epsilon_K^L$ for the model problem.

K	$L = K - 1$	$K - 2$	$K - 3$	$K - 4$	$K - 5$	$K - 6$
5	$3.70 \cdot 10^{-5}$	$1.77 \cdot 10^{-4}$				
6	$3.79 \cdot 10^{-6}$	$5.94 \cdot 10^{-5}$	$*1.06 \cdot 10^{-4}$			
7	$5.65 \cdot 10^{-7}$	$3.55 \cdot 10^{-6}$	$9.55 \cdot 10^{-6}$	$1.61 \cdot 10^{-5}$		
8		$5.55 \cdot 10^{-7}$	$1.10 \cdot 10^{-7}$	$*1.95 \cdot 10^{-6}$	$3.62 \cdot 10^{-6}$	
9			$4.62 \cdot 10^{-7}$	$6.17 \cdot 10^{-7}$	$1.05 \cdot 10^{-6}$	$2.02 \cdot 10^{-6}$
10				$6.08 \cdot 10^{-8}$	$*8.31 \cdot 10^{-8}$	$1.22 \cdot 10^{-7}$
11					$1.87 \cdot 10^{-8}$	$2.37 \cdot 10^{-8}$

consists of $O(\sqrt{n})$ nodes. The table clearly shows that with the presented fast evaluation algorithm, the multisummation can be performed on a grid with $O(\sqrt{n})$ points at negligible loss of accuracy.

To get a better insight into the error introduced by the fast evaluation, we also monitor the *incremental* error, defined by the l_2 -norm of the difference in the solution on level K when direct summation is performed on level $L + 1$ and when direct summation is performed on level L :

$$I\epsilon_K^L = \left(\frac{1}{n} \sum_{i=0}^n |(G^h \tilde{u}^h)^{K,L+1}(\mathbf{x}_i^h) - (G^h \tilde{u}^h)^{K,L}(\mathbf{x}_i^h)|^2 \right)^{1/2}. \quad (48)$$

This quantity measures the additional error introduced by transferring the evaluation from level $L + 1$ to level L . The incremental errors are listed in Table 4. The table shows that the incremental evaluation errors are in all relevant cases of the same order of magnitude as the discretization error.

To determine the computational complexity of the fast evaluation, the expended operations are counted. The operation-count is obtained as the sum of the transfer-costs and the cost of the coarsest grid multisummation (of both the original transform and the corrections) for all grids involved in the fast evaluation. For the results in Table 3, the computational work per grid h node is displayed in Table 5. The leftmost column of the table shows the costs of direct summation. It can be

TABLE 5
Work per gridpoint spent in the evaluation of $S^{h,(2,2)}$ for the model problem.

K	L=K	K-1	K-2	K-3	K-4	K-5	K-6
5	1089	185	151				
6	4225	450	70	*74			
7	$1.7 \cdot 10^4$	1368	120	56	71		
8	$6.6 \cdot 10^4$	4743	351	63	*44	53	
9	$2.6 \cdot 10^5$		1197	97	27	24	27
10	$1.1 \cdot 10^6$			309	43	*24	24
11	$4.4 \cdot 10^6$				94	21	16

seen that these costs amount to $O(n^2)$ operations. The entries marked by asterisks are the computational costs in case direct multisummation is performed on a grid with $O(\sqrt{n})$ nodes. As expected, the costs per grid point decrease for increasing K . It is anticipated that only the costs of the transfers on the fine grids and of the coarsest grid multisummation remain as $h \rightarrow 0$. The total number of operations is then $W \approx 2\bar{p} + 1$. In the present case this yields $W \approx 9$. The results in table 5 suggest that this may indeed be obtained for sufficiently large K .

7. CONCLUSION

Motivated by the demand for local grid refinement techniques in practical applications, this work examined the extension to multiple dimensions of a new algorithm for the fast evaluation of integral transforms with asymptotically smooth kernels. The discretization procedure was outlined. Details were presented for the fast evaluation method in the instance of multiple dimensions. It was shown that the asymptotic work estimates in [8] can indeed be obtained, provided that multilevel evaluation of the corrections is applied. The softened kernels in the fast evaluation algorithm were constructed by applying the softening operation sequentially with respect to each coordinate direction. The optimization of softening parameters for multidimensional transforms was discussed.

The fast evaluation algorithm was tested for a two dimensional model problem. The results showed that with the new algorithm the evaluation of multidimensional transforms is also more efficient than with previous algorithms. Moreover, the results confirmed the expected asymptotic work estimates for the considered test case.

REFERENCES

1. A.W. Appel. An efficient program for many-body simulation. *SIAM J. Sci. Stat. Comput.*, 6:85–103, 1985.
2. D. Bai and A. Brandt. Local mesh refinement multilevel techniques. *SIAM J. Sci. Stat. Comput.*, 8:109–134, 1986.
3. J. Barnes and P. Hut. A hierarchical $O(n \ln n)$ force calculation algorithm. *Nature*, 324:446–449, 1986.
4. A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comp.*, 31:333–390, 1977.
5. A. Brandt. Guide to multigrid development. In W. Hackbusch and U. Trottenberg, editors, *Multigrid Methods (Proc. Köln-Porz, 1981)*, number 960 in Lecture Notes in Math. 960, pages 220–312, New York, 1982. Springer Verlag. also included in: GMD Studien No. 85, Gesellschaft für Mathematik und Datenverarbeitung MBH, Bonn, Germany (1984).
6. A. Brandt and A. A. Lubrecht. Multilevel matrix multiplication and the fast solution of integral equations. *J. Comput. Phys.*, 90:348–370, 1990.
7. A. Brandt and C.H. Venner. Fast evaluation of integral transforms on adaptive grids. In W. Hackbusch and G. Wittum, editors, *Multigrid Methods V (Proc. Stuttgart, 1996)*, volume 3 of *Lecture Notes in Computational Science and Engineering*, pages 20–44, Berlin, 1998. Springer Verlag. also appeared as Internal Report Carl F. Gauss Minerva center for scientific computation, WI/CG-5.
8. A. Brandt and C.H. Venner. Multilevel evaluation of integral transforms with asymptotically smooth kernels. *SIAM J. Sci. Stat. Comput.*, 19:468–492, 1998.
9. M.M. Grigoriev and G.F. Dargush. A fast multi-level boundary-element method for the helmholtz equation. *Comput. Methods Appl. Mech. Engrg.*, 193:165–203, 2004.
10. M.M. Grigoriev and G.F. Dargush. A multi-level boundary-element method for two-dimensional steady heat diffusion. *Num. Heat Transf., Part B*, 46:329–356, 2004.
11. Z.P. Nowak and W. Hackbusch. On the complexity of the panel method. In *International conference on Modern Problems in Numerical Analysis*, Moscow, 1986.
12. L. Reichel. A fast method for solving certain integral equations of the first kind with application to conformal mapping. *J. Comput. Appl. Math.*, 14:125–142, 1986.
13. C.H. Venner and A.A. Lubrecht. *Multilevel Methods in Lubrication*, volume 73 of *Tribology Series*. Elsevier, Amsterdam, 2000.
14. P. Wesseling. *An introduction to multigrid methods*. Pure and Applied Mathematics. Wiley, Chichester, 1992.

APPENDIX A

In many cases, if a kernel consists of a summation of components, the smoothness of the kernel with respect to a variable is dictated by a single component. One can then define a so-called principal smoothness component: Let $\Theta = \{y-x | x \in \Theta, y \in \Theta\}$. If $G^l(t) = \sum_i G_i^l(t)$ is asymptotically smooth in t_k and there exist an index j , a $\bar{t} \in \Theta$ and a minimal order $\bar{p} \in \mathbb{N}$, such that for all $t \in \Theta$, all orders $p \geq \bar{p}$ and all indices i it holds that:

$$|G_i^{l-p\mathbf{e}_k}(t)| \leq \alpha |G_j^{l-p\mathbf{e}_k}(\bar{t} + (t_k - \bar{t}_k)\mathbf{e}_k)|, \quad (\text{A.1})$$

for some positive constant α , then $\bar{G}_k^l(\eta) \equiv \alpha G_j^l(\bar{t} + (\eta - \bar{t}_k)\mathbf{e}_k)$ is the principal smoothness component of $G^l(t)$ in the k -direction. Because the smoothness of the kernel in t_k is essentially governed by the principal smoothness component, suitable softening parameters, m_k and p_k , can be conveniently determined from the properties of \bar{G}_k^l .

If the mesh width h is sufficiently small, then it is generally possible to employ $m_k = 0$ and a fixed, minimal order of transfer, \bar{p}_k , depending only on l_k , during several of the first coarsening stages. This is a result of the use of integrated kernels. However, this does not apply if the evaluation is a subtask in the solution of the integral equations, in which case it is necessary to use $m_k = O(\bar{p}_k)$ on the finest grids to maintain the stability properties of the single-grid operator. For the isolated evaluation of the integral transform, [8] shows that $m_k = 0$ requires $\bar{p}_k > l_k$. Due to the singularity in the original kernel, the integrated kernel, $G^l(x, y)$, contains components with singular or discontinuous derivatives at $y_k = x_k$ ($t_k = 0$). Consequently, in order to determine the left-hand side of (42), it is necessary to distinguish between the region $\Theta_s = \{t \in \Theta \mid |t_k| \leq \bar{p}_k H_k/2\}$, i.e. the region where the singularity is in the interpolation interval, and the region $\Theta \setminus \Theta_s$. In $\Theta \setminus \Theta_s$, the kernel $G^l(x, y)$ is \bar{p}_k times differentiable with respect to x_k and y_k . In this case the average order \bar{p}_k interpolation error satisfies

$$\|G_H^l - \mathbb{I}_H^{H-H_k\mathbf{e}_k/2} G_H^l\|_{1, \Theta \setminus \Theta_s} \leq (\gamma_2 H_k)^{\bar{p}_k} \|G^{l(\bar{p}_k\mathbf{e}_k)}\|_{1, \Theta \setminus \Theta_s}. \quad (\text{A.2})$$

Using (A.1),

$$\|G^{l(\bar{p}_k\mathbf{e}_k)}\|_{1, \Theta \setminus \Theta_s} \leq \|\bar{G}_k^{l(\bar{p}_k)}\|_{1, \Theta \setminus \Theta_s}. \quad (\text{A.3})$$

Because $\bar{G}_k^{l(\bar{p}_k-1)}$ vanishes asymptotically at infinity, the right-hand side of (A.3) is approximately $|\bar{G}_k^{l(\bar{p}_k-1)}(\bar{p}_k H_k/2)|$. In Θ_s , however, equation (A.2) is useless as a result of the singular derivatives. The interpolation error in this region is bounded by the local value of the component with singular derivatives. Taylor expansion of the kernel in the neighborhood of the singularity yields that for sufficiently small t_k the behavior of this component is dominated by the principal smoothness component. Hence, the interpolation error for $0 \leq |y_k - x_k| \leq p_k H_k/2$ is bounded by

$$\|G_H^l - \mathbb{I}_H^{H-H_k\mathbf{e}_k/2} G_H^l\|_{1, \Theta \setminus \Theta_s} \leq \|\bar{G}_k^l\|_{\max, \Theta_s}. \quad (\text{A.4})$$

Generally, the principal smoothness component is a monotonic function. The right-hand side of (A.4) is then $|\bar{G}_k^l(\bar{p}_k H_k/2)|$. Notice that the contribution of the region Θ_s to the entire integral is just $O(\bar{p}_k H_k)$. By (42) and (A.2) through (A.4),

it is anticipated that $m_k = 0$ can indeed be used on grids where the following two requirements are satisfied:

$$|\bar{G}_k^l(\bar{p}_k H_k/2)| \bar{p}_k H_k \leq h_k^{s_k-d(s_k-l_k)} c_{G,u} c_h c_\gamma, \quad (\text{A.5})$$

$$|\bar{G}_k^{l(\bar{p}_k-1)}(\bar{p}_k H_k/2)| H_k^{\bar{p}_k} \leq h_k^{s_k-d(s_k-l_k)} \gamma_2^{-\bar{p}_k} c_{G,u} c_h c_\gamma. \quad (\text{A.6})$$

Next, we investigate $m_k > 0$ and any $p_k \geq \bar{p}_k$. The softened kernel is $p_k - 1$ times continuously differentiable in the k -direction. Hence, the p_k -order interpolation error is bounded by

$$\|G_{\mathbf{H}}^l - \mathbb{I}_{\mathbf{H}}^{H-H_k \mathbf{e}_k/2} G_{\mathbf{H}}^l\|_{1,\Theta} \leq (\gamma_2 H_k)^{p_k} \|G_{\mathbf{H}}^{l(p_k \mathbf{e}_k)}\|_{1,\Theta}. \quad (\text{A.7})$$

The derivative of the softened kernel in (A.7) is dominated by the derivative of its softened principal smoothness component:

$$\|G_{\mathbf{H}}^{l(p_k \mathbf{e}_k)}\|_{1,\Theta} \leq \alpha_3 \|\bar{G}_{\mathbf{H}k}^{l(p_k)}\|_{1,\Theta}, \quad (\text{A.8})$$

for some positive constant α_3 . Due to the locality of the softening operation, it is necessary to distinguish two regions, viz., the region $\Theta_s = \{\mathbf{t} \in \Theta \mid |t_k| \leq (m_k + p_k/2) H_k\}$ where the softening domain is in the interpolation interval and the interpolation error is determined by the softening polynomial, and the region $\Theta \setminus \Theta_s$, where the error results from the interpolation of the original kernel. In $\Theta \setminus \Theta_s$, the average order p_k interpolation error is bounded by

$$\|G_{\mathbf{H}}^l - \mathbb{I}_{\mathbf{H}}^{H-H_k \mathbf{e}_k/2} G_{\mathbf{H}}^l\|_{1,\Theta \setminus \Theta_s} \leq (\gamma_2 H_k)^{p_k} \|\bar{G}_k^{l(p_k)}\|_{1,\Theta \setminus \Theta_s}. \quad (\text{A.9})$$

The right-hand side of (A.9) is approximately $|\bar{G}_k^{l(p_k-1)}(m_k H_k)|$. In the region Θ_s , the p_k -derivative of the softening polynomial determines the interpolation error. By (27) and (28), for even ($odd = 0$) and odd ($odd = 1$) functions \bar{G}_k^l , the p_k -derivative of the softened principal smoothness component reads:

$$\bar{G}_{\mathbf{H}k}^{l(p_k)}(\eta) = \sum_{i=0}^{p_k-1} \sum_{j=0}^{p_k-1} B_{ij}^{-1} b(p_k, i) (m_k H_k)^{-(2i+odd-j)} \eta^{(2i+odd-p_k)} \bar{G}_k^{l(j)}(m_k H_k), \quad (\text{A.10})$$

for $\eta \in [0, m_k H_k]$, with

$$b(i, j) = \begin{cases} \frac{(2j+odd)!}{(2j+odd-i)!} & 2j+odd \geq i, \\ 0 & \text{otherwise,} \end{cases} \quad (\text{A.11})$$

and \mathbf{B} the $(p_k \times p_k)$ -matrix with entries $B_{ij} = b(i, j)$ ($i, j = 0, \dots, p_k - 1$). The right-hand side of (A.10) can be used to construct a convenient, sharp bound of the form:

$$\alpha_3 \|\bar{G}_{\mathbf{H}k}^{l(p_k)}\|_{1,\Theta_s} \leq f(p_k, l_k) F(m_k H_k) \quad (\text{A.12})$$

with F some elementary positive function. The contribution of Θ_s to the left-hand side of equation (42) is just $O(m_k H_k)$. By (42) and (A.7) through (A.12), one

arrives at the following two requirements for p_k and m_k :

$$\left| \bar{G}_k^{l(p_k-1)}(m_k H_k) \right| (H_k)^{p_k} \leq h_k^{s_k-d(s_k-l_k)} c_{G,u} c_h c_\gamma \gamma_2^{-p_k}, \quad (\text{A.13})$$

$$f(p_k, l_k) m_k H_k^{p_k+1} F(m_k H_k) \leq h_k^{s_k-d(s_k-l_k)} c_{G,u} c_h c_\gamma \gamma_2^{-p_k}. \quad (\text{A.14})$$

Summarizing, whenever H_k satisfies (A.5) and (A.6), $m_k = 0$ and $p_k = \bar{p}_k$ is used. Otherwise, m_k and p_k must satisfy (A.13) and (A.14).

APPENDIX B

To obtain the optimal transfer parameters for the fast evaluation of the model problem, the principal smoothness components of the kernel (11) is derived first. Because of the symmetry of $G^{(2,2)}(\mathbf{t})$ in t_1 and t_2 , it is sufficient to obtain the component in one direction. Observing that $G_0^{(2,2)}(\mathbf{t})$ in (29) can be recast into

$$G_0^{(2,2)}(\mathbf{t}) = \frac{1}{2} t_1^2 t_2 \ln \left(t_2 + \sqrt{t_1^2 + t_2^2} \right) - \frac{1}{2} t_1^2 t_2 \ln |t_1|, \quad (\text{B.1})$$

analysis of the derivatives of the components reveals that the smoothness of $G^{(2,2)}(\mathbf{t})$ in the 1-direction is dominated by the second term in (B.1). Assuming that the dimensions of the domains $\underline{\Omega}$ and Ω are $O(1)$, the principal smoothness component of $G^{(2,2)}(\mathbf{t})$ is

$$\bar{G}_k^{(2,2)}(\eta) = \eta^2 \ln |\eta|. \quad (\text{B.2})$$

In general, the principal smoothness component in the k -direction of $G^l(\mathbf{t})$ for the family of kernels with $G(\mathbf{x}, \mathbf{y})$ by (4) is

$$\bar{G}_k^l(\eta) = \eta^{l_k} \ln |\eta|. \quad (\text{B.3})$$

Hence, it is useful to maintain a general notation. Because the principal smoothness component (B.3) is identical to the integrated kernel in the 1-dimensional model-problem treated in [8], one can consult [8] for details of the below optimization procedure.

By (A.5), (A.6) and (B.3), it is anticipated that $\bar{p}_k = l_k + 2$ and $m_k = 0$, i.e. no softening at all, can be used on all grids with mesh size H_k satisfying

$$H_k^{l_k+1} \ln(H_k) = O \left(h^{s_k-d(s_k-l_k)} \right). \quad (\text{B.4})$$

Next, consider $m_k > 0$. For all $p_k \geq l_k + 2$, requirement (A.14) is more restrictive than requirement (A.13) and, consequently, m_k and p_k can be determined from the minimization of (43) subject to (A.14). From [8],

$$\left\| \bar{G}_{H_k}^{l(p_k)} \right\|_{1, \Theta_s} \leq f(p_k, l_k) (m_k H_k)^{l_k-p_k}, \quad (\text{B.5})$$

with $f(p, l) = (2(p-l)/e)^{p-l}$. Requirement (A.14) then assumes the following form:

$$(m_k/\gamma_2)^{l_k-p_k+1} f(p_k, l_k) \leq g, \quad (\text{B.6})$$

where

$$g = \frac{h^{s_k - d(s_k - l_k)}}{H_k^{l_k + 1}} c_{G,u} c_h c_\gamma \gamma_2^{-(l_k + 1)}. \quad (\text{B.7})$$

Assuming equality in (B.6),

$$m_k = \gamma_2 \exp \left(\frac{\ln(g) - \ln(f(p_k, l_k))}{l_k - p_k + 1} \right). \quad (\text{B.8})$$

Subsequently, (B.8) is substituted in (43) and the p_k for which $dW^H/dp_k = 0$ is calculated. Making minor simplifications such as $(p_k - l_k)/(p_k - l_k - 1) \approx 1$, it follows that W^H is minimized when

$$\frac{32}{3} \gamma_2 \bar{H}^{-1} e^{1/\chi} = \chi, \quad (\text{B.9})$$

where

$$\chi = \frac{-(p_k - l_k - 1)}{(p_k - l_k - 1) + \ln(g)}. \quad (\text{B.10})$$

Summarizing, for the evaluation of the model transform, the optimal value of p_k for the transfer of the grid $H/2$ softened transform to grid H is the maximum of the lowest non-negative integer satisfying

$$p_k \geq \frac{-\chi}{\chi + 1} \ln(g) + l_k + 1 \quad (\text{B.11})$$

and $p_k \geq l_k + 2$. The corresponding optimal value of m_k is the first integer that satisfies:

$$m_k \geq \frac{3}{32} \bar{H} \gamma_2^{-1} \chi (p_k - l_k - 1). \quad (\text{B.12})$$

Notice that $\ln(g)$ can be conveniently rewritten as

$$\ln(g) = c_a + (s_k - d(s_k - l_k)) \ln(h) - (l_k + 1) \ln(H_k), \quad (\text{B.13})$$

where c_a is a constant depending on the geometry of the interpolation, the order of discretization, the average kernel value in Θ and the derivatives of $u(\mathbf{y})$. Decreasing c_a causes p_k and m_k to increase, so that c_a controls the accuracy of the fast evaluation.